viewpoints

Kevin Fu and Wenyuan Xu

Inside Risks
Risks of Trusting the Physics of Sensors

Protecting the Internet of Things with embedded security.

SENSORS ARE TRANSDUCERS that translate the physical into the electrical. Computer software then interprets and operates on the binary representations rather than the direct physical or electrical quantities. For instance, drone software uses the abstraction of a signed integer to represent the output of a gyroscope for flight stability and attitude control.[4] A *transduction attack* exploits a vulnerability in the physics of a sensor to manipulate its output or induce intentional errors. For example, malicious acoustic interference can influence the output of sensors trusted by software in systems ranging from smartphones to medical devices to autonomous vehicles. Autonomous systems should remain trustworthy despite untrustworthy components. Techniques from embedded security can help protect against analog threats to autonomous systems in the Internet of Things.

Threats. Thieves can break into cars using man-in-the-middle (MITM) attacks against keyless entry systems.[5] Automotive manufacturers can neutralize MITM attacks with proper use of cryptography. However, these MITM attacks exploit automotive systems that intend for radio waves to allow access. In contrast, transduction attacks use unintended functions of circuitry to threaten the integrity and availability of sensor output. Cryptography will not suffice to defend against transduction attacks. Attackers can exploit the physics of materials to fool sensors into becoming unintentional receivers of unwanted, malicious signals. The threat has grown such that the U.S. government warns manufacturers of transduction attacks that exploit physics of sensors.[1]

Sensors face two types of analog threats: opportunistic attacks requiring no special-purpose equipment, and advanced attacks that require special-purpose transmitters and basic understanding of physics. For instance, an opportunistic attack could use phishing to trick a person into playing untrustworthy music videos on a smartphone. The sound waves can influence the output of an accelerometer.[14] Because a smartphone includes both a speaker and accelerometer, the adversary needs no transmitter or spe-

cial equipment to carry out this attack. An advanced attacker may build custom acoustic or radio frequency emitters. For instance, an adversary could use a Long-Range Acoustic Device (LRAD) to deliver intense sound waves from a mile away.

Vulnerabilities. Billions of deployed sensors lack designed-in protections against intentional physical manipulation.[4,12–15] Most likely, the sensors were designed before the community understood the security risks. Researchers have repeatedly shown how an adversary can not only cause denial of service, but also control the sensor output itself with malicious analog signals at the resonant frequency of the sensor. Vulnerabilities tend to lurk deep within the physics of analog sensors. The risks bubble up into the software layer.

The DolphinAttack[15] represents a transduction attack vulnerability whereby inaudible sounds can trick speech recognition systems into executing phantom commands. Microphones, especially miniature microelectromechanical systems (MEMS) microphones, can hear ultrasound. Although the circuits and software attempt to at-

tenuate such high-frequency sounds, an adversary can inject fake voice commands with ultrasound. The ultrasonic method exploits non-linear behavior within the signal path conditioning of the circuitry. The microphone is tricked into functioning as an unintentional acoustic demodulator. The DolphinAttack can silently manipulate almost all popular speech recognition systems, such as Siri, Google Now, Samsung S Voice, Huawei HiVoice, Cortana, Alexa, and the voice-controlled navigation system in an Audi automobile.

Malicious Back-Door Coupling. In the context of aircraft safety, front-door interference refers to unwanted signals that enter a system directly via an antenna port whereas back-door interference refers to unwanted signals that enter a system indirectly by coupling to its wires and other components.[9] A transduction attack can use malicious back-door coupling to cause sensors to function as unintentional receivers and demodulators. That is, a sensor designed to sense one phenomenon (for example, deceleration of a car) may also accept unwanted signals (for example, sound waves at the resonant frequency of the sensor) without distinguishing the sources. Malicious back-door coupling can exploit a resonant frequency of unremarkable amplitude to overshadow a legitimate signal. There are many examples of malicious back-door coupling to violate sensor integrity. Malicious back-door radio waves tricked pacemakers into disabling pacing shocks.[4] Malicious interference blending both front-door and back-door coupling fooled Tesla's sensors into hiding and spoofing obstacles,[7] as shown in the three-image series in this column depicting real, spoofed, and jammed distances.

A hacker does not necessarily require special-purpose equipment to exploit back-door coupling in sensors. One could co-opt nearby software-controlled emitters common in laptop computers, smartphones, speaker systems, and even light bulbs. For instance, our research demonstrated how playing sounds embedded in a YouTube video allows an adversary to control the output of a smartphone's MEMS accelerometer. The exploit works because of mechanical coupling between the sensor and the smart-

Advanced sensor attacks. Sensors translate the physical into the electrical for interpretation by a computer system. However, analog signals can spoof data by exploiting the physics of sensors. This photo shows how malicious electromagnetic waves can trick software processing signals from a thermocouple into displaying an impossibly low temperature (−1409° F is 527° K below absolute zero).
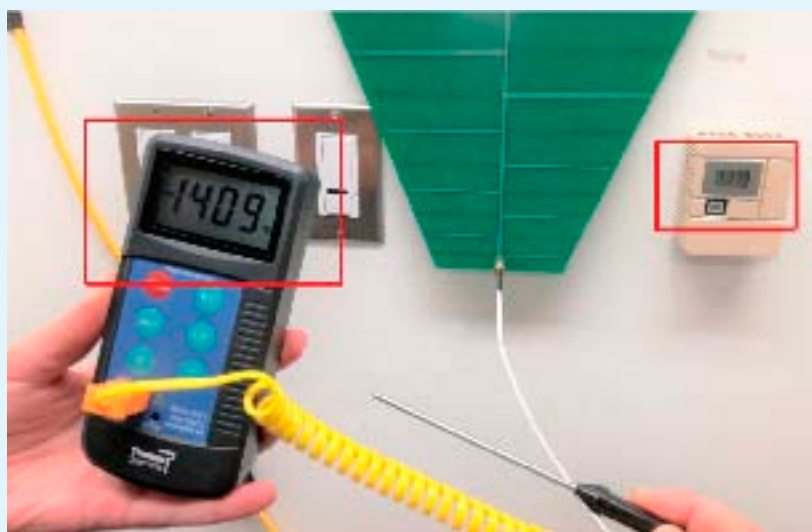
phone's built-in speaker that emits malicious signals modulated over a carrier at the resonant frequency of the sensor to induce a chosen sensor output.[14]

Trustworthy Embedded Systems

Protecting against transduction attacks is difficult because the consequences arise as software symptoms, but the risks begin in the physics of hardware. Good security practices such as static analysis, fuzz testing, and signed software updates are insufficient to protect against a sensor delivering intentionally false data. Software security tools were not designed to control for analog security risks. Thus, we recommend a return to classic engineering approaches for more trustworthy embedded systems to cope with threats to the underlying physics of sensor technology.

▸ Shift from component-centric security to system-centric tolerance of untrustworthy components.

▸ Make the output of sensor hardware more continuously checkable by software for adversarial influence.

▸ Make attacks more difficult by manufacturing circuits in a manner to reduce effects of resonance.

Avoid component-centric security. Sensor systems should remain safe despite adversarial influence on untrustworthy components. Fault-

tolerant systems pioneered the non-adversarial variant of this problem by limiting damage with techniques such as compartmentalization. However, faults and defects that develop after verification cannot be detected by verification. In computer security, the adversary controls the probability distribution of maliciously induced errors in components and can induce faults after verification.

Systems that treat security as just another component rather than a property will survive poorly against analog adversaries who can manipulate sensors with transduction attacks. Trusted components do not suffice to ensure a trustworthy system. For instance, a secure processor will happily sign false sensor data if blindly accepting output from a trusted sensor rather than continuously doubting and checking trustworthiness of sensor output. Trustworthy components can fail catastrophically when attacks succeed; trustworthy systems can fail more gracefully when attacks succeed. Key to overall system trustworthiness is the ability for systems to check the trustworthiness of sensor output.

Make the security of sensor output continuously checkable. A central principle of information security[8] is to consider inputs as circumspect until shown trustworthy (for example, by

Malicious interference fooled Tesla's sensors into hiding and spoofing obstacles:[7] (a) Real distance; (b) spoofed distance; (c) jammed distance.

(a)

(b)

(c)

satisfying an independent check). Sensors may contain self-calibration circuits tested with injected signals during manufacture or power-up to verify the sensors perform as specified. Self-checking is difficult even when mother nature is the adversary. NOAA discovered its algorithms erroneously excluded output from a temperature sensor in Alaska because of a false positive of an anomaly detection algorithm.[3] Sensors threatened by intentional transduction attacks must clear an even higher bar of continuous checkability.

Sensor interfaces could continuously convey additional evidence for applications to perform end-to-end checks of sensor trustworthiness. Some sensors already maintain debugging information internally, but do not expose the information across the hardware-software API. Sensors could expose spectral analytics, confidence indicators, or other hints such that software applications could better detect threats such as signals at known resonant frequencies. A system can also compare data from multiple sensors operating on different physical principles (for example, comparing a reed switch and hall-effect sensor that sense magnetic fields). An engineering challenge is reconciling security with constraints of performance, board space, and cost. Exposing checkable hints of sensor output trustworthiness would enable a shift away from component-centric security toward system-centric security.

Specify physical security. When we reported an acoustic security flaw that allowed adversarial influence of accel-

erometer outputs, one manufacturer made an innovative recommendation that specifies how to more securely attach a sensor to a circuit board.[2] The response to the CERT report may be the first example of advising customers to physically manipulate a drill bit, rather than digital bits, to mitigate a security vulnerability. Customers were advised to use inner mounting posts to a hard case to reduce board deflection near a sensor and ensure the vibrations of the board are above the resonant frequency of the sensor. Drilling holes differently in a circuit board can shift the resonant frequency out of the range that nearby acoustic transducers can generate or that the sensor's non-linearities can demodulate. The manufacturer also advised customers to place physical trenches around boards containing speakers to reduce mechanical coupling. Such simple, physical approaches can serve as effective compensating controls to decrease the risk of transduction attacks.

> Autonomous systems should remain trustworthy despite untrustworthy components.

Embedded Security Education
Security is a system property. Thus, design of a sensor-driven, safety-critical system requires supervision by a systems engineer with broad knowledge of computer security risks. Team leaders for such systems will need to master skills from physics, electrical engineering, and mechanical engineering to computer science, information science, public policy, and ethics.

Interdisciplinary teams. For medical devices and vehicles, an engineering team will minimally need a blend of experts from mechanical engineering, electrical engineering, and computer science who share an awareness of risks and recognize the value of working together. Students destined for solving these types of problems need early exposure to interdisciplinary teamwork in classes and internships. However, not all engineers must master the underlying physics of computer security. Instead, every team member needs a basic awareness of the risks. A system always includes risks that will fall outside an individual team member's area of expertise. Thus, each engineer has an ethical responsibility to maintain awareness of analog security risks, inform management of uncontrolled risks, and know when to ask for expert help from a team leader.

The notion of interdisciplinary education is not new to computer science. In the 1990s, the software engineering community debated a shift toward interdisciplinary education beyond the confines of computer science.[10,11] Similarly, a good engineer for embedded security will not simply be a good

computer scientist or a good programmer. Interdisciplinary education and teamwork is key to ensuring security of sensor-driven, safety-critical systems.

Educational opportunities for embedded security. Aspiring system-security engineers need opportunities to learn fundamentals of embedded security. However, computer science curricula have little room to add material given the pressure to meet the industry's demand for gifted programmers. How can computer science programs create expert embedded security graduates under these constraints? Computer science cannot succeed alone.

Engineering schools should offer interdisciplinary educational programs for ambitious students to learn how to protect cyberphysical systems. Students would learn not just fundamentals of computer science and computer security, but also the physics of computational abstractions. A software engineer may take computer security courses to learn threat modeling, cryptography, and secure programming methodologies. To master the concepts and skills for embedded security, an engineer would also take courses that teach the fundamentals of signals and systems, communication theory, and classical physics. For instance, defending against transduction attacks involves spectral analysis, mechanical resonance, and modulation. Students wishing to become experts in embedded security must understand how each layer of computation from sensors to human behavior can fail when subjected to adversarial interference.

Back to basics. Students are losing an appreciation for the physical machines that implement computational abstractions. Students graduating from departments that diminish the role of computing machinery will not be prepared to create trustworthy cyberphysical systems. For instance, students unaware of transduction attacks may falsely believe that verified software is failure-proof. Math-centric departments tend to avoid courses that emphasize building physical systems. If a department eliminates computer architecture, students may seek comfort hiding behind a beautiful Java facade rather than facing the ugly limitations of computing machinery. Even engineering-centric computer science departments succumb to this problem. Students may desire immediately marketable programming skills over understanding the fundamental limitations of the machines on which their software runs.

Students creating the next generation of trustworthy cyberphysical systems need an exposure to the physical limitations of the machines implementing each abstraction. An effective way to do this is to include labs featuring experiments of the kinds suggested earlier in this column. Tomorrow's software engineer must master both math-centric and engineering-centric skills while understanding the physical limitations of computational machinery.

Conclusion

Sensors are vulnerable to spoofing by transduction attacks. Cyberphysical systems must cope with analog threats that an adversary could exploit without any special-purpose equipment. Automobiles decide whether to deploy an airbag based on data from accelerometers.[14] Pacemakers and defibrillators decide whether to emit shocks based on data from cardiac sensors.[6] It is inevitable and predictable that hackers will try to manipulate sensors to cause havoc. Autonomous systems making safety-critical decisions should remain safe even when an adversary can exploit physics to influence the output of sensors. The community can reduce these risks by designing sensors to be continuously checkable for security properties and by increasing opportunities for students to master the physics of computer security and principles of embedded security.

> Cyberphysical systems must cope with analog threats that an adversary could exploit without any special-purpose equipment.

References
1. Alert (ICS-ALERT-17-073-01A). MEMS Accelerometer Hardware Design Flaws (Update A), (Apr. 11, 2017); http://bit.ly/2CjTdcD.
2. Analog Devices Advisory to ICS-ALERT-17-073-01 (Apr. 2017); http://bit.ly/2EPF9cc.
3. Arndt, D. Alaskan North Slope climate change just outran one of our tools to measure it. (Dec. 6, 2017); http://bit.ly/2AFNBJz.
4. Foo, D. et al. Ghost Talk: Mitigating EMI signal injection attacks against analog sensors. In *Proceedings of IEEE Symposium on Security and Privacy* (Oakland, CA), May 2013.
5. Francillon, A., Danev, B., and Capkun, S. Relay attacks on passive keyless entry and start systems in modern cars. In *Proceedings of Network and Distributed System Security Symposium (NDSS)*, The Internet Society, 2011.
6. Fu, K. Pacemaker recall exposes national need for research and education in embedded security. In *Computing Community Consortium (CCC)*, (Sept. 2017); http://bit.ly/2xBEqel.
7. Liu, J., Yan, C., and Xu, W. Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicles. In DEFCON24 (Aug. 2016); http://bit.ly/2EQNOLs.
8. Neumann, P.G. Fundamental trustworthiness principles. In *New Solutions for Cybersecurity*. In *MIT Press/Connection Science*, H. Shrobe, D. Shrier, A. Pentland, Eds., Cambridge, MA, 2018.
9. Nguyen, T. Cumulative interference to aircraft radios from multiple portable electronic devices. In *IEEE Conference on Digital Avionics Systems*, 2005.
10. Parnas, D.L. Education for computing professionals. In *IEEE Computer 23*, 1 (Jan. 1990), 17–22.
11. Parnas, D.L. Software engineering programmes are not computer science programmes. In *Annals of Software Engineering 6* (1998), 19–37. (Reprinted in IEEE Software (Nov./Dec. 1999), 19–30.
12. Rouf, I. et al. Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study. In *Proceedings of USENIX Security Symposium*, (Aug. 2010).
13. Son, S. Rocking drones with intentional sound noise on gyroscopic sensors. In *Proceedings of USENIX Security Symposium* (Aug. 2015).
14. T. Trippel, T. et al. WALNUT: Waging doubt on the integrity of MEMS accelerometers with acoustic injection attacks. In *Proceedings of IEEE European Symposium on Security and Privacy* (Euro S&P), (Apr. 2017), http://bit.ly/2Cl2KQn.
15. Zhang, G. et al. DolphinAttack: Inaudible voice commands. In *Proceedings of ACM Conference on Computer and Communications Security (CCS)*, October 2017.

Kevin Fu (kevinfu@umich.edu) is Associate Professor of Electrical Engineering and Computer Science at the University of Michigan.

Wenyuan Xu (wyxu@zju.edu.cn) is Professor and Chair of the Department of Systems Science and Engineering at Zhejiang University.